Remarks

Claims 1-24 were pending in the subject application. By this Amendment, the applicants have amended claim 4 to correct an informality. No new subject matter has been added. Favorable consideration of the claims now presented, in view of the remarks set forth herein, is earnestly solicited.

Claims 1-24 have been rejected under 35 U.S.C. §103(a) as being unpatentable over Yazdani *et al.* (U.S. Patent No. 6,859,455) in view of Rajasekaran *et al.* (U.S. Pat. Pub. No. 2002/0124003). The applicants respectfully traverse this rejection.

Attached, for the Examiner's convenience is a Declaration under 37 CFR §1.132 by one of the inventors, Dr. Sartaj Kumar Sahni, explaining differences between Yadzani's patent and the data structures of the current application.

Considering independent claims 1 and 19, first, in contrast to the assertion at page 3 of the Action that Yazdani *et al.* disclose a method comprising applying a top level balanced binary search tree with at least one lower level range search tree to a nonintersecting highest priority rule table (citing Figure 4; col. 12, lines 37-38; Figure 11b; reference 100 (range 01-04); and Figure 30), Yazdani *et al.* do not disclose or suggest a data structure with a top level binary search tree and a lower level range search tree. In fact, as explained by Dr. Sahni at point 6, "the binary search tree of Figure 4 is given as an example of a search that fails to find the longest matching prefix (see col. 15, lines 3-14). The binary search tree shown in Figure 4 is not a balanced binary search tree (PTST). Rather, each node of the binary search tree shown in Figure 4 stores a prefix and not a point value as specified by the claims."

Furthermore, as explained by Dr. Sahni at point 7:

> The term "level" is used in Yazdani *et al.* to refer to the distance from the tree root, i.e., the tree root is at level 0, its children are at level 1, their children at level 2, and so forth. In contrast, the claimed top level balanced binary search tree (PTST) has at least one node comprising a lower level range search tree (RST). This context refers to an embedding level. The entire tree constructed by each of Yazdani's methods is a top-level tree. None of the trees disclosed by Yazdani include lower-level embedded trees or other embedded structures.

Rajasekaran *et al.* does not cure this defect.

Next, Figure 11b of Yazdani *et al.* simply shows how partitions can be subdivided when building a tree for ranges. According to Yazdani *et al.*, as illustrated by this range decomposition, it is possible for a range to be stored in multiple nodes. In addition, with respect to the claimed limitation "applying a range allocation rule to allocate to each PTST node a subset of the nonintersecting ranges and corresponding priorities," Yazdani *et al.* at col. 6, lines 51-62 discloses a division method that applies only to prefixes. String length has no meaning in the context of range. Thus, this disclosed division method of Yazdani *et al.* is not a range allocation rule. Furthermore, the range allocation rule used by Yazdani *et al.* is capable of storing the same range in many nodes. For example, referring to Figure 11b of Yazdani *et al.*, O4 is stored in 3 nodes. In contrast, the subject claimed range allocation rule allocates to each PTST node a subset of the nonintersecting ranges and corresponding priorities. That is, each range is stored in one node.

In addition, in contrast to the indication at page 3 of the Office Action that Yazdani *et al.* discloses applying an RST to organize the subset at col. 6, lines 63-67 and col. 7, lines 1-4, this cite does not apply to RSTs. It is not clear from the Action how an m-way tree can be considered the claimed range search tree.

Therefore, not only does Yazdani *et al.* fail to disclose a balanced binary search tree, rules for search, insert and delete, empty nodes and range search list as admitted at page 4 of the Action, Yazdani *et al.* do not teach or suggest two-level (top-level and lower-level) structures or a method to handle ranges that avoids creating multiple copies of the same range. As pointed out by the Amendment submitted June 8, 2007 and again by Dr. Sahni at point 10 of the attached Declaration, it is impossible to convert the Yazdani *et al.* structures to balanced structures that have good search/insert/delete complexity. Furthermore, Rajasekaran *et al.* do not disclose any balanced structure or method to search/insert/delete in logarithmic time. The methods disclosed by Rajasekaran *et al.* employ well-known techniques to search/insert/delete using m-way tries (see also the Declaration of Dr. Sahni at points 16-17). It should be noted that (log log n) is not the same as $\log^2 n$; (log log n) is log (log n) while $\log^2 n$ is (log n) * (log n). Furthermore, referring to paragraph [0009] of the Rajasekaran *et al.* reference, it is apparent that the complexity of the disclosed structure assumes a restricted alphabet set and "require substantial memory resources to perform the search, and as a result are not very cost effective when the data set to be searched is large." Practical

balanced structures for exact match queries have complexity O(log n) for search/insert/delete. These are not readily adapted to longest prefix matching or to any of the other problems addressed by the structures claimed in the subject patent application.

In addition, a pointer being null is not the same as a node being empty as claimed in subject claim 1. All pointer-based structures have some number of null pointers. For example, every leaf in a binary tree has 2 null pointers. Furthermore, paragraph [0059] of the Rajasekaran *et al.* reference makes no mention of a range search tree or even of a range. Neither the basic m-way trie nor the variant shown in Figure 9 are range search trees and neither can handle ranges. The tries of Rajasekaran *et al.* are not balanced, and Rajasekaran *et al.* show no rules for search, insert, or delete for a balanced structure. Instead Rajasekaran *et al.* merely show rules for search, insert, and delete for their m-way tries, which are not balanced. (See the Declaration of Dr. Sahni at points 16-17)

<u>Considering independent claims 7, 13, 21, and 23</u>, as stated above, Yazdani *et al.* do not teach or suggest a top level balanced binary search tree (PTST) having another search structure embedded in a node; the other search structures including the RST, an array linear list (ALL), and a W-bit vector (bit(z)) as claimed. The fundamental differences between the claimed structures and the structures disclosed by Yazdani *et al.* do not permit achieving one set of structures from the other set using balancing or other transformational strategies.

The applicants appreciate that the Examiner provided comments in the Response to Amendment section following the rejection of claims 1-24. However, the applicants respectfully object to the characterization of the arguments submitted on June 8, 2007. Not only do Yazdani *et al.* fail to teach or suggest "binary search tree and array linear, ALL, or W-bit vector," the structures of Yazdani *et al.* do not allow insert/delete actions to occur in O(log n) time. As stated in Dr. Sahni's Declaration at point 8, "it is impossible to insert or delete into/from this structure in less than linear (O(n)) time . . . it is impossible to insert/delete from the Yazdani structures in O(log n) time." The data structures of the current patent application employ <u>binary search trees</u> (such as red-black search trees) in a very unique manner for <u>dynamic</u> routing processes. Specifically, a binary search tree of <u>points</u> at the top level is applied, wherein each node of the top-level binary search tree contains either another search tree (such as a lower level binary range search tree), an array linear list (ALL), or W-bit vector. The lower level search tree, ALL, or W-bit vector contain a subset of

ranges/prefixes in the router table. To obtain the O($\log^2 n$) search and O(log n) insert/delete complexities as claimed, the top level balanced binary search tree can retain a limited number of empty nodes in the structure. As disclosed in the application, by accounting any empty nodes, at most 2n nodes are needed in the top-level binary search tree to enable the claimed complexities (*i.e.*, O($\log^2 n$) rule/prefix matching and O(log n) insertion/deletion). Without the capacity to include empty nodes in the top-level binary search tree (PTST), it is not possible to get the claimed complexities. Thus, it is not possible to get efficient insert/delete complexities (O(log n) time) using the structures of Yazdani or Turner or a combination thereof. These same shortcomings are found in the combination of Yadzani and Rajasekaran.

Accordingly, the applicants respectfully request reconsideration and withdrawal of the rejection of claims 1-24 under 35 U.S.C. §103(a).

In view of the foregoing remarks, the applicants believe that the currently pending claims are in condition for allowance, and such action is respectfully requested.

The Commissioner is hereby authorized to charge any fees under 37 CFR §§1.16 or 1.17 as required by this paper to Deposit Account No. 19-0065.

The applicants also invite the Examiner to call the undersigned if clarification is needed on any of this response, or if the Examiner believes a telephone interview would expedite the prosecution of the subject application to completion.

Respectfully submitted,

Sarah Knight

Sarah J. Knight
Patent Attorney
Registration No. 58,722
Phone: 352-375-8100
Fax No.: 352-372-5800
Address: P.O. Box 142950
Gainesville, FL 32614-2950

SJK/la

Attachment: Declaration by Sartaj Kumar Sahni, Ph.D with the following attachments:

1) Exhibit A (*Curriculum vitae*)
2) Excerpts from "Fundamental of Data Structures" (Computer Science Press, Maryland, 1976)